

BUNDESREPUBLIK DEUTSCHLAND



PRIORITY DOCUMENT
CERTIFIED COPY OF

Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

Aktenzeichen: 100 55 168.8
Anmeldetag: 08. November 2000
Anmelder/Inhaber: Siemens Aktiengesellschaft, München/DE
Bezeichnung: Industrielle Steuerung auf der Basis verteilbarer
Technologischer Objekte
Priorität: 03.08.2000 DE 100 37 990.7
03.08.2000 DE 100 37 971.0
IPC: G 05 B 19/04

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ur-springlichen Unterlagen dieser Patentanmeldung.

München, den 04. Januar 2001
Deutsches Patent- und Markenamt

Der Präsident

Im Auftrag

Reust

Beschreibung

Industrielle Steuerung auf der Basis verteilbarer Technologischer Objekte

5

Die Erfindung bezieht sich auf eine industrielle Steuerung für technische Prozesse, insbesondere für Produktionsmaschinen.

10 Ferner bezieht sich die Erfindung auf ein Verfahren zur Programmierung bzw. Projektierung von industriellen Steuerungen für technische Prozesse, insbesondere für Produktionsmaschinen.

15 Eine industrielle Steuerung kann dabei ein eigenes Gerät sein, sie kann aber auch in einen Computer, einem PC, einem eigenständigen Gerät oder einem Antrieb integriert sein.

20 Bisher bekannte industrielle Steuerungen zur Automatisierung technischer Prozesse basieren im Wesentlichen entweder auf einer "SPS-Funktionalität", einer "MC-Funktionalität" oder auf einer Technologiefunktionalität. Da im Rahmen solcher Funktionalitäten ein gewisser Funktionsumfang fest vorgeschrieben ist, ist eine optimale Anpassung an die Anforderungen eines speziellen Prozesses häufig nur bedingt möglich, wobei im konkreten Anwendungsfall oft eine ganze Gruppe von Funktionen überflüssig ist (z.B. ist beim Einsatz einer MC-Steuerung für Werkzeugmaschinen, evtl. vorhandene Funktionalität für Verpackungsmaschinen überflüssig).

25

Aus der DE 197 40 550 ist außerdem eine Vorrichtung zum Steuern eines technischen Prozesses und/oder zur Steuerung der Bewegung einer Verarbeitungsmaschine bekannt, die ein Steuerprogramm abarbeitet. Dieses Steuerprogramm besteht aus einer

35 Vielzahl von Softwaremodulen. Prozesssteuerungsfunktionalitäten von an sich bekannten speicherprogrammierbaren Steuerungen und Bewegungsfunktionalitäten von an sich bekannten MC-

Steuerungen sind in einem einheitlichen, konfigurierbaren Steuerungssystem verwirklicht. Die einzelnen Software-Module werden hier jedoch durch jeweils eine Teilsteuering abgearbeitet, so dass für jedes Software-Modul eine zentrale Recheneinheit vorzusehen ist.

Weiterhin ist aus der DE 198 53 205 ein Verfahren zur Steuerung technischer Prozesse bekannt, das auf einer Instanziierbarkeit sowie einer bedarfsgerechten Verschaltung von Software-Komponenten mit vorgebbarer, zumindest parametrierbarer Funktionalität basiert. Die Verschaltung und die Projektierung der Software-Komponenten erfolgt hierbei allerdings noch nicht optimal.

Der Erfindung liegt daher die Aufgabe zugrunde, für jeweils unterschiedliche Steuerungsaufgaben und unterschiedliche Randbedingungen bzw. Anforderungen des zugrunde liegenden technischen Prozesses in einfacher Weise optimale Ausprägungen einer industriellen Steuerung sowohl hinsichtlich ihrer Steuerungsstruktur als auch hinsichtlich ihrer Funktionalität zu erstellen.

Die Erfinder sind dabei von der Erkenntnis ausgegangen, dass das Runtime- und/oder Engineering-System der industriellen Steuerung sowohl SPS- als auch Bewegungs- und/oder Technologie-Funktionalität bedient und dass durch die Möglichkeit des dynamischen Zuladens von Funktionscode in das Runtime- und/oder Engineering-System der industriellen Steuerung jeweils eine optimale Ausprägung, d.h. Skalierung der Steuerung möglich sein müsste, wobei außerdem durch eine Trennung von technologischer Funktionalität und Gerätefunktionalität die Erstellung bzw. Projektierung der Steuerung erleichtert werden würde.

Gemäß der Erfindung wird die oben genannte Aufgabe für eine industrielle Steuerung der eingangs genannten Art dadurch gelöst, dass die Steuerung ein allgemein einsetzbares, vorzugs-

weise technologieneutrales, Basissystem für die Steuerungsgrundfunktionalität aufweist, wobei instanziierbare Technologieobjekttypen die Grundfunktionalität der Steuerung um technologische Funktionalitäten ergänzen und nach einer vom Anwender zuschneidbaren Instanziierung als Technologische Objekte in seinen jeweiligen Applikationen zur Verfügung stehen, wobei eine Trennung zwischen technologischer Funktionalität und Gerätefunktionalität erfolgt.

Ein Technologisches Objekt repräsentiert vorzugsweise eine Komponente der realen Welt. Im Kontext industrieller Steuerungen können dies z.B. Komponenten von Werkzeugmaschinen oder Produktionsmaschinen sein. Die Technologischen Objekte stellen eine definierte technologische, abgeschlossene Funktionalität bereit. Sie können untereinander verschaltet werden, um komplexe technologische Aufgaben zu realisieren. Dadurch dass die technologische Funktionalität der Steuerung durch Technologische Objekte, die vorzugsweise reale Komponenten, gebildet wird, ist einem Anwender oder Nutzer der Steuerung die technologische Mächtigkeit, d.h. die Fähigkeit der Steuerung sofort transparent. Als softwaretechnologische Einheit kann ein Technologisches Objekt außerdem von einem Anwender sehr leicht in unterschiedlichen Applikationen und Steuerungen wiederverwendet werden. Ein Anwender kann bei der Nutzung von Technologischen Objekten von deren Implementierung abstrahieren. Vom Anwender in seinen Anwendungsprogrammen direkt einsetzbare Technologische Objekte entstehen durch ihre Instanzierung aus Technologieobjekttypen. Aus einem einmal definierten Technologieobjekttyp können beliebig viele (zugeschneiderte) Instanzen von Technologischen Objekten gewonnen werden. Dadurch dass die Instanzierung sowohl im Engineering-System als auch im Runtime-System erfolgen kann, ist es für einen Anwender sehr leicht und sehr komfortabel möglich, die Technologischen Objekte in seinen Anwendungen zu verwenden. Die funktionale Mächtigkeit einer Steuerung ist somit sehr leicht erweiterbar. Die Erweiterbarkeit wird le-

diglich durch HW-Restriktionen (z.B. CPU-Leistung oder Speicherbeschränkungen) begrenzt.

Weiterhin hat der Anwender die Möglichkeit das vorhandene Basissystem für die Steuerungsgrundfunktionalität um solche Funktionalitäten zu erweitern, die er wirklich für seine Anwendungen benötigt. Dies geschieht dadurch, dass er explizit bestimmte benötigte Technologische Objekte zum Basissystem der Steuerung hinzulädt. Ein Anwender kann sich somit individuell eine Steuerung mit einer bestimmten Funktionalität beschaffen. Üblicherweise in Steuerungen vorhandene nichtbenötigte Funktionalitäten werden dadurch vermieden und verursachen somit keinen Overhead.

Ein weiterer Vorteil liegt in der Trennung von technologischer Funktionalität und Gerätefunktionalität. In den Technologischen Objekten wird von den Geräten abstrahiert, auf denen die Technologischen Objekte ablaufen. Somit kann sehr leicht die Zuordnung eines Technologischen Objektes auf ein Gerät geändert werden und die Programmerstellung (d.h. die Nutzung der Technologischen Objekte in den Anwenderprogrammen) kann unabhängig von den Geräten erfolgen. Die Geräte selber stellen somit nur die Ablaufumgebung für die Technologischen Objekte dar. Die tatsächliche Zuordnung von Technologischen Objekten zu Geräten kann der Anwender in einer für ihn optimalen Art und Weise vornehmen. Optimierungskriterien sind z.B. Auslastung, räumliche Verteilung oder die Buslänge.

Außerdem liegt ein Vorteil in der Entwicklung und in der Produktion solcher skalierbaren Steuerungen. Steuerungen, die mit einer notwendigen Grundfunktionalität (Basissystem) ausgeliefert werden, lassen sich in großer Stückzahl sehr einfach herstellen (economies of scale).

Eine erste vorteilhafte Ausgestaltung der Erfindung liegt darin, dass eine automatische Generierung bzw. Projektierung von Kommunikationsverbindungen zwischen Technologischen Ob-

5 jekten basierend auf der zugrundeliegenden Hardware-Topologie und/oder der technologischen Lösung erfolgt. Im Engineering-
system werden die Zuordnungsinformationen von Technologischen Objekten zu Geräten, die Geräte- und Netzttopologie, sowie die und das Datenvolumen ausgewertet und daraus die automatische Projektierung der Kommunikationskanäle erzeugt. Damit wird für einen Anwender die Programmerstellung erleichtert.

10 Eine weitere vorteilhafte Ausgestaltung der Erfindung liegt darin, dass bei der automatischen Generierung bzw. Projektierung der Kommunikationsverbindungen zwischen Technologischen Objekten von den Technologischen Objekten zugewiesene oder erworbene Qualitätsattribute berücksichtigt werden. Diese automatische Kommunikationsprojektierung ermöglicht eine effiziente Nutzung der eingesetzten Geräte- und Netzttopologie, da dabei abstrakte "Quality of Service"-Anforderungen wie z.B. Broadcast, Taktsynchronität oder Übertragungszeit optimal auf die Geräte- und Buseigenschaften abgebildet werden. Im Engi-
neeringssystem werden die Zuordnungsinformationen von Techno-
15 logischen Objekten zu Geräten, die Geräte- und Netzttopologie, sowie die "Quality of Service"-Anforderungen und das Datenvolu-
lumen ausgewertet und daraus die automatische Projektierung der Kommunikationskanäle erzeugt.

20 25 Eine weitere vorteilhafte Ausgestaltung der Erfindung liegt darin, eine flexible Verschiebbarkeit und/oder Verteilbarkeit der Technologischen Objekte auf unterschiedlich oder gleich performante Hardware-Systeme und/oder Laufzeitsysteme er-
folgt. Technologische Objekte sind plattform- bzw. hardware-
unabhängig. Sie enthalten keine plattform- bzw. hardware-
spezifischen Eigenschaften und können somit sehr leicht auf unterschiedliche Hardware-Systeme und/oder Laufzeitsysteme geladen, verschoben und verteilt werden. Durch die Möglich-
keit der Verschiebbarkeit und der Verteilbarkeit der Techno-
30 35 logischen Objekte auf unterschiedlich oder gleich performante Hardware-Systeme kann ein Anwender Technologische Objekte sehr flexibel benutzen und einsetzen. Er muss sich bei der

Verschiebung und der Verteilung der Technologischen Objekte nicht um Restriktionen bezüglich der zugrundeliegenden Performance der Hardware- und/oder von Laufzeit-Systemen kümmern. Weiterhin ist es durch die Verschiebung und Verteilung 5 der Technologischen Objekte möglich, die Last in einem System flexibel zu verteilen und zu balanzieren.

Eine weitere vorteilhafte Ausgestaltung der zugrunde liegenden Erfindung liegt darin, dass eine flexible Verschiebbarkeit und/oder Verteilbarkeit der Technologischen Objekte auf unterschiedlich oder gleich performante Hardware- und/oder Laufzeit-Systeme innerhalb eines Projektes erfolgt, wobei 10 sich ein Projekt auf Daten und/oder Programme von einer oder mehreren Steuerungseinheiten bezieht. Ein Anwender hat somit 15 die Möglichkeit innerhalb eines Projektes Geräte unterschiedlicher Hardware einzusetzen, die auch unterschiedlich performant sein können, auf die er Technologische Objekte leicht und flexibel verteilen kann, ohne die unterschiedliche Performance der Geräte berücksichtigen zu müssen.

20 Eine weitere vorteilhafte Ausgestaltung der zugrunde liegenden Erfindung liegt darin, dass die Verteilung der Funktionalität der Technologischen Objekte auf miteinander in Echtzeit durch taktsynchron äquidistant kommunizierende Steuerungseinheiten erfolgt. Die Technologischen Objekte können somit auf 25 Geräte bzw. Steuerungseinheiten verteilt werden, die über ein Kommunikationsmedium in Verbindung stehen, das eine taktsynchrone äquidistante Kommunikation erlaubt. Somit können die Technologischen Objekte in Echtzeit miteinander kommunizieren. In einem Projekt sind die Instanzen von Technologischen 30 Objekttypen eindeutig referenzierbar und können (HW-) plattformübergreifend genutzt werden.

35 Eine weitere vorteilhafte Ausgestaltung der vorliegenden Erfindung liegt darin, dass eine technologische Skalierung hinsichtlich der Funktionalität der Steuerung durch die Zuladbarkeit von Technologieobjekttypen erfolgt. Damit hat der An-

wender die Möglichkeit eine funktionale Skalierung seiner Steuerung zu erreichen. Er kann somit sehr einfach die Funktionalität der Steuerung an die zugrundeliegenden und vorliegenden Bedürfnisse und Randbedingungen anpassen. Die Erweiterbarkeit bezieht sich sowohl auf Gerätefunktionalität als auch auf technologische Funktionalität.

Eine weitere vorteilhafte Ausgestaltung der vorliegenden Erfindung liegt darin, dass eine Verschaltung der Technologischen Objekte zu komplexen Technologischen Objekten, sog. Container-Objekten erfolgt. Dadurch hat der Anwender die Möglichkeit aus "einfachen" Technologischen Objekten komplexe Technologische Objekte zu erstellen, die im Vergleich zu den "einfachen" Technologischen Objekten eine höherwertige bzw. komplexere technologische Funktionalität repräsentieren und zur Verfügung stellen. Die Verschaltung erfolgt durch hierarchische Beziehungen zwischen den Technologischen Objekten und/oder Datenflussbeziehungen.

Eine weitere vorteilhafte Ausgestaltung der Erfindung liegt darin, dass einem Anwender unterschiedliche Sichten auf die Technologischen Objekte zur Verfügung stehen. Die Abstraktionsmechanismen, die die Technologischen Objekte zur Verfügung stellen, erlauben (je nach Anwendungsphase oder Anwendertyp) unterschiedliche Sichten auf sie. Aus dem Engineering gibt es z.B. eine Projektsicht (üblicherweise in Form einer Baumdarstellung) und/oder eine Inbetriebnahmesicht (z.B. für das Anlegen und Konfigurieren der Instanzen). Aber es steht auch eine programmiertechnische Sicht zur Verfügung. In dieser Sicht werden dem Anwender z.B. Methoden und Attribute der Technologischen Objekte zur Verfügung gestellt. Aus ergonomischen Gesichtspunkten werden die Sichten einem Anwender in Form von grafischen Benutzeroberflächen zur Verfügung gestellt, z.B. als Icons und/oder Masken.

Eine weitere vorteilhafte Ausgestaltung der vorliegenden Erfindung liegt darin, dass eine rückwirkungsfreie Programmie-

rung eines Technologischen Objektes bezüglich der anderen vorhandenen Technologischen Objekte und des Steuerungsbasis-
systems, sofern nicht explizit eine Rückwirkung programmiert
bzw. projektiert ist, vorgesehen ist. Der Anwender kann somit
5 das Verhalten eines Technologischen Objektes unabhängig von
Rückwirkungen anderer Technologischen Objekte oder des Steue-
rungsbasisystems programmieren. Wenn erforderlich oder ge-
wünscht kann er aber explizit eine Rückwirkung programmieren
bzw. projektierten. Die Flexibilität des Anwenders bei der
10 Programmierung Technologischer Objekte wird dadurch erhöht.

Eine weitere vorteilhafte Ausgestaltung der vorliegenden Er-
findung liegt darin, dass die Darstellung der Technologischen
Objekte im Engineering-System durch grafische Elemente
15 und/oder Masken erfolgt. Durch diese grafische Benutzerober-
fläche wird der Anwender beim Gebrauch der Technologischen
Objekte unterstützt. Produktivität und Qualität der Projek-
tierung bzw. der Programmierung werden dadurch erhöht.

20 Eine weitere vorteilhafte Ausgestaltung der vorliegenden Er-
findung liegt darin, dass die Technologieobjekttypen zu Tech-
nologischen Paketen zusammengefasst werden. Technologische
Pakete stellen eine Clusterung von technologisch und/oder
funktional zusammengehörenden Technologieobjekttypen dar.

25 Durch das Hinzuladen von Technologischen Paketen zum Basis-
system einer Steuerung, können Steuerungen mit jeweils dedi-
zierten Funktionsumfang erhalten werden. Solche Steuerungen
haben wenig funktionalen Overhead. Durch die Clusterung und
Zuordnung von Technologieobjekttypen zu Technologiepaketen
30 wird zum einen eine Strukturierung und Klassifizierung er-
reicht und zum anderen sind die Technologiepakete ein geeig-
netes Mittel, um die Technologieobjekttypen auf das Runtime-
System einer Steuerung zu laden.

35 Gemäß der Erfindung wird die oben genannte Aufgabe für ein
Verfahren der eingangs genannten Art durch die folgenden auf-
einander folgenden Schritte gelöst:

- a) Verwendung eines Basissystems mit einer vorzugsweise technologieneutralen Grundfunktionalität,
- b) Instanziierung der Technologischen Objekte,
- c) Verschaltung der Technologischen Objekte zu Technologi-
5 schen Objekten komplexer Funktionalität,
- d) Verteilung und/oder Platzierung der Technologischen Ob-
jekte auf die Geräte,
- e) automatische Generierung der Kommunikationskanäle zwi-
schen den Technologischen Objekten,
- 10 f) Wiederverwendung insbesondere von komplexen bereits ver-
schalteten Technologischen Objekten in anderen Projekten.

Dadurch hat der Anwender die Möglichkeit auf eine systemati-
sche und folgerichtige Weise die Funktionalität einer ge-
wünschten Steuerung zu erreichen, wobei sichergestellt ist,
15 dass die erhaltene Steuerung so gut wie keinen funktionalen
Overhead beinhaltet. Ein weiterer Vorteil besteht darin, dass
ein Anwender bei der Erstellung der Anwenderprogramme die
Technologischen Objekte unabhängig von der Hardware und den
20 Geräten, auf denen sie letztendlich ablaufen, verwenden kann.
Nach der Instanziierung und der Verschaltung der Technologi-
schen Objekte erfolgt erst ihre Verteilung auf die Hardware
bzw. die Geräte. Die Zuordnung zu den Geräten kann jederzeit
geändert werden. Es liegt somit eine strenge Trennung von
25 technologischer Funktionalität und Gerätelfunktionalität vor.
Die technologische Funktionalität der Technologischen Objekte
ist unabhängig von der Gerätelfunktionalität, d.h. von den Ge-
räten, auf denen sie ablaufen. Die Geräte selber stellen nur
die Ablaufumgebung für die Technologische Objekte dar. Tech-
30 nologische Objekte (einfache und/oder komplexe und/oder ver-
schaltete lassen sich daher sehr leicht in anderen Projekten
wiederverwenden. Die automatische Generierung der Kommunika-
tionskanäle zwischen den Technologischen Objekten (automati-
sche Kommunikationsprojektierung) ermöglicht eine effiziente
35 Nutzung der eingesetzten Gerät- und Netztopologie und unter-
stützt den Anwender bei der Projektierung bzw. der Programm-
erstellung.

Eine weitere vorteilhafte Ausgestaltung der vorliegenden Erfindung liegt darin, dass bei der Generierung der Kommunikationskanäle Qualitätsattribute der Technologischen Objekte eingehalten werden. Durch die Berücksichtigung vom Anwender
5 vorgebbarer Qualitätsanforderungen (z.B. Übertragungszeit, Taktsynchronizität, Broadcast) wird die Geräte- und Netztopologie noch effizienter genutzt und der Anwender braucht bei der Projektierung bzw. Programmierung der Kommunikationskanäle nur Qualitätsattribute als Input für die automatische Ge-
10 nerierung der Kommunikationskanäle angeben.

Eine weitere vorteilhafte Ausgestaltung der vorliegenden Erfindung liegt darin, dass die Schritte b) und e) optional erfolgen. Dadurch, dass Technologische Objekte nicht zwangsläufig verschaltet und/oder in anderen Projekten wiederverwendet zu werden brauchen, wird die Flexibilität für den Anwender erhöht.
15

Die wesentlichen, mit der Erfindung erzielten Vorteile bestehen also insbesondere darin, dass ein Anwender direkt technologische Funktionalität in seinen Anwendungen verwenden kann, die ihm durch Technologische Objekte, die Elemente der realen Welt entsprechen, in einer für ihn adäquaten Weise zur Verfügung gestellt werden und dass für einen Anwender eine strenge
20 Trennung von technologischer Funktionalität und Gerätefunktionalität vorliegt. Geräte stellen nur die Ablaufumgebung für Technologische Objekte dar. Die technologische Funktionalität der Technologischen Objekte ist unabhängig von der Gerätefunktionalität.
25

Ein weiterer Vorteil liegt darin, dass die Funktionalität von industriellen Steuerungen sozusagen "plug and play"-mäßig de-diziert erweiterbar ist. Auf diese Weise wird eine technologische Skalierung der Steuerung erreicht.
30

35 Ein Ausführungsbeispiel der Erfindung ist in der Zeichnung dargestellt und wird im folgenden erläutert.

Dabei zeigen:

FIG 1 in einem Strukturbild ein Engineering-System, das zugehörige Runtime-System und den zu steuernden technischen Prozess,

FIG 2 zeigt in einem Übersichtsbild wie ein Anwenderprogramm auf technologische Funktionalität im Runtime-System zugreift,

FIG 3 zeigt in einer abstrakten Schemadarstellung ein Technologisches Objekt mit Anwenderschnittstelle,

FIG 4 zeigt in Form eines sog. Verschaltungsdigrammes Technologische Objekte, die einen Gleichlaufverbund darstellen,

FIG 5 zeigt einen Gleichlaufverbund mit Umschaltmöglichkeiten zwischen verschiedenen Leitwertquellen und Gleichlaufgesetzen, ebenfalls in Form eines Verschaltungsdigrammes,

FIG 6 zeigt in Form eines Verschaltungsdigrammes die Ver- schaltung des Technologischen Objektes Messtaster,

FIG 7 zeigt in Form eines Verschaltungsdigrammes die Ver- schaltung des Technologischen Objektes Nocke,

FIG 8 zeigt in Form eines Verschaltungsdigrammes Verschal- tungen mit Gleichlauftechnologieobjekten,

FIG 9 zeigt ebenfalls in Form eines Verschaltungsdigrammes die Zuordnung eines Technologischen Objektes Kurvenscheibe zum mehreren Gleichlaufobjekten,

FIG 10 zeigt in einem Übersichtsbild die Clusterung von Technologieobjekttypen zu einem Technologiepaket und

5 FIG 11 zeigt in einem Übersichtsbild die Kommunikationsstruktur zwischen zwei Geräten.

In der Darstellung gemäß FIG 1 wird in Form eines Strukturbildes gezeigt, dass die Steuerung eines technischen Prozesses P über mindestens ein Runtime-System RTS1-RTS3 von industriellen Steuerungen erfolgt. Die Verbindung zwischen den 10 Runtime-Systemen RTS1-RTS3 der Steuerung und dem technischen Prozess P geschieht bidirektional über Ein-/Ausgänge EA1-EA3. Die Programmierung der Steuerung und damit das Festlegen des Verhaltens der Runtime-Systeme RTS1-RTS3 geschieht im Engineering-System ES. Das Engineering-System ES enthält Werkzeuge für die Konfigurierung, Projektierung und Programmierung für Maschinen bzw. für die Steuerung technischer Prozesse. Die im Engineering-System ES erstellten Programme werden 15 über die Informationspfade I1-I3 jeweils in die Runtime-Systeme RTS1-RTS3 der Steuerungen übertragen. Durch die drei Punkte ist angedeutet, dass weitere Steuerungen und Runtime-Systeme vorhanden sein können. Bezüglich seiner Hardware-Ausstattung besteht ein Engineering-System ES üblicherweise aus einem Computersystem mit Graphikbildschirm (z.B. Display), Eingabehilfsmitteln (z.B. Tastatur und Maus), Prozessor, Arbeits- und Sekundärsspeicher, eine Einrichtung für die Aufnahme computerlesbarer Medien (z.B. Disketten, CD's) sowie Anschlusseinheiten für einen Datenaustausch mit anderen Systemen (z.B. weiteren Computersystemen, weitere Steuerungen 20 für technische Prozesse) oder Medien (z.B. Internet). Eine Steuerung besteht üblicherweise aus Eingabe- und Ausgabeeinheiten, sowie aus Prozessor und Programmspeicher.

Darstellung gemäß FIG 2 zeigt zwei Runtime-Systeme RTS4 und 35 RTS5 von industriellen Steuerungen, dargestellt als Rechteck. Die Runtime-Systeme RTS4 und RTS5 enthalten jeweils einen UMC-Kernel UMC-K, sowie die Technologischen Objekte TO1 bis

TO_n, wobei die jeweiligen UMC-Kernels als auch die Technologischen Objekte unterschiedlich sein können, die Technologischen Objekte auch in ihrer Anzahl. Der UMC-Kernel UMC-K stellt das Basissystem der Steuerung dar, dieses Basissystem
5 enthält die Grundfunktionalität der Steuerung. Der UMC-Kernel UMC-K ist in rechtwinkliger Stufenform dargestellt. Zu ihm können Technologische Objekte T_{O1} bis T_{On} hinzugeladen werden. Durch dieses Hinzuladen wird der Funktionsumfang des Basissystems erweitert. Die Technologischen Objekte T_{O1} bis T_{On}
10 sind als Rechtecke dargestellt, durch ihre Anordnung in FIG 2 wird angedeutet, dass sie den UMC-Kernel UMC-K erweitern.
Durch die drei Punkte wird angedeutet, dass ein bis mehrere
Technologische Objekte T_{O1} bis T_{On} hinzugeladen werden können und somit eine technologische Skalierung der gesamten Steue-
15 rung erreicht wird. Zentriert am oberen Rand von FIG 2 ist das Anwenderprogramm AP1 in Form einer schematischen Papier- fahne dargestellt. Durch die Zugriffspfeile ZGP1 bis ZGP4 ist dargestellt, dass ein Anwender in seinem Anwenderprogramm AP1 direkt auf Funktionalitäten des UMC-Kernels UMC-K aber auch
20 auf Funktionalitäten der Technologischen Objekte T_{O1} bis T_{On} zugreifen kann, sowohl von RTS4 als auch von RTS5 oder von einem weiteren Runtime-System (ebenfalls angedeutet durch drei Punkte). Diese angebotene Funktionalitäten der Runtime- Systeme RTS4 und RTS5 (oder von weiteren Runtime-Systemen)
25 kann ein Anwender direkt in seinem Anwendungsprogramm AP1 verwenden.

Zur Präzisierung: Zur Erweiterung des Basissystems eines Runtime-Systems werden Technologieobjekte üblicherweise in Form
30 von Technologieobjekttypen hinzugeladen. Solche Technologie- objekttypen sind z.B. Achsen, Nocken, Kurvenscheiben oder ähnliches. Technologieobjekttypen sind instanziierbar. Ein Anwender verwendet in seinen Anwendungsprogrammen AP1 für konkrete Applikationen Instanzen von Technologieobjekttypen.
35 Solche Instanzen sind dann projektweit eindeutig definiert und identifizierbar. Die direkte Verwendung von zugeladenen Technologischen Objekten in Anwenderprogrammen AP1 als je-

weils eigenständige Programmobjekte wäre prinzipiell auch denkbar, ist aber für einen Anwender für die Programmerstellung unflexibel.

5 Darstellung gemäß FIG 3 zeigt in einer abstrakten Schemadarstellung die Anwendersicht eines Technologischen Objektes, d.h. einer Instanz eines Technologieobjekttyps. Diese Spezifikation eines Technologischen Objektes TOS wird als Rechteck dargestellt, das aus fünf Teilen besteht. Der oberste erste
10 Teil, abgetrennt von den folgenden Teilen durch eine durchgezogene Linie, enthält den Typ des zugrunde liegenden Technologischen Objektes (TO-Type) und den TO-Identifier, d.h. die projekteindeutige Bezeichnung der Instanziierung. Der nächst folgende Teil enthält die Konfigurationsdaten (Configuration Data) mit den Konfigurationsvariablen <configuration variable_1> bis <configuration variable_n>. Über die Konfigurationsdaten wird das Technologische Objekt in seiner grundsätzlichen Wirkungsweise eingestellt. Die Konfigurationsdaten werden über das Engineering-System (ES; FIG 1) eingestellt
15 und können optional über Zugriffsfunktionen aus dem Anwendungsprogramm (AP1; FIG 2 und AP2, AP3; FIG 11) heraus gelesen bzw. geschrieben werden. In der Darstellung gemäß FIG 3 werden die Konfigurationsdaten durch eine gestrichelte Linie von den Systemvariablen (System Data) abgetrennt. Die Systemvariablen
20 <system variable_1> bis <system variable_m> sind aus dem Anwendungsprogramm (AP1; FIG 2 und AP2, AP3; FIG 11) heraus veränderbar und wie Programmvariable nutzbar. Systemvariablen können lesbar oder les-/schreibbar sein. Durch die Systemvariablen werden außerdem die Zustände von Technologischen Objekten repräsentiert. Zustandsübergänge können durch Ereignisse und/oder Befehle ausgelöst werden. Über die Konfigurationsdaten und die Systemvariablen erfolgt die Parametrierung
25 der Technologischen Objekte. Der nächste Abschnitt sind die Befehle (Commands), von den Systemvariablen ebenfalls durch eine gestrichelte Linie getrennt. Die Befehle <command_1> bis <command_xy> stellen aufrufbare Funktionen dar, die die Funktionalität eines Technologischen Objekts repräsentieren. Die-

se Funktionen haben definierte Bezeichner, Funktionsparameter und lokale Werte. Die Funktionen können Parameter besitzen. Beim Aufruf von Funktionen können optionale Parameter weglassen werden, hierfür werden dann Defaultwerte eingesetzt.

5 Zusätzlich zur technologischen Funktionalität besitzt ein Technologisches Objekt aber auch Befehle, die das Grundverhalten des Technologischen Objektes bestimmen, z.B.

- Befehl zum Rücksetzen in einem definierten Ausgangszustand
- Befehl um einen anstehenden Fehler gezielt rückzusetzen
- 10 - Befehle um in den Simulationsbetrieb zu setzen und rückzusetzen (Im Simulationsbetrieb erfolgt ein Durchlauf des Programms ohne konkrete Ausgabe an die Aktoren, bzw. Einlesen von den Sensoren).
- Befehle um das Technologische Objekt aktiv/inaktiv zu setzen
- 15 - Auskunftsfunctionen.

Der nächste Abschnitt der Spezifikation eines Technologischen Objektes TOS sind die Alarne (alarms). In FIG 3 sind die Alarne durch eine gestrichelte Linie von den Befehlen abgetrennt.

20 Die Darstellung gemäß FIG 3 enthält die Alarne <alarm_1> bis <alarm_k>. Ein Technologisches Objekt hat Überwachungen und kann im Fehlerfall definierte Alarne, gegebenenfalls mit Alarminformationen und vordefinierten Reaktionen absetzen. Die technologischen Alarne werden am Technologischen Objekt festgestellt bzw. erzeugt. Technologische Alarne

25 haben eine technologieobjekttypspezifisch eingestellte Reaktion, z.B. Bewegungsstopp (die möglichen Reaktionen sind technologieobjekttypspezifisch und daher bei den einzelnen Technologieobjekttypen explizit beschrieben). Weiterhin besitzen die technologischen Alarne einen technologieobjekttypspezifischen Identifikator (z.B. Alarmnummer) und Parameter.

30 Sie besitzen somit ein einstellbares Verhalten auf die Programmbearbeitung (globale Reaktion) und erlauben weiterhin für jeden Fehler instanzspezifische Einstellungen und Reaktionen, die bei der Inbetriebnahme am Engineering-System (ES; FIG 1) vorgenommen werden.

Ein Anwender kann Befehle von Technologischen Objekten synchron oder asynchron nutzen, je nach Einstellung. Dadurch kann ein Befehl sowohl zyklisch geschrieben (üblich bei einer Speicherprogrammierbaren Speicherung) aber auch ereignisgesteuert (üblich bei Bewegungssteuerungen) programmiert werden. Im synchronen Modus bleibt z.B. das technologische Objekt, das einen Positionierbefehl ausführt, solange in seinem Zustand, bis das Positionierziel erreicht wurde. Im asynchronen Modus dagegen läuft zeitgleich zur Ausführung des Positionierbefehls das Technologische Objekt in seinem Programmablauf weiter und kann dabei andere Zustände einnehmen. Das Technologische Objekt kann dann z.B. durch Polling geprüft werden, ob das Positionierziel erreicht wurde.

Die Darstellung gemäß FIG 4 zeigt als Verschaltungsdiagramm die Verschaltung des Technologischen Objektes "Gleichlauf" GL1 mit anderen Technologischen Objekten. Die Technologischen Objekte werden dabei als doppelt umrandete Rechtecke dargestellt, bei denen die jeweils zusammen gehörenden Ecken durch eine Verbindungsleitung verbunden sind. Durch die Verschaltung des Technologischen Objektes "Gleichlauf" GL1 mit den Technologischen Objekten "Leitachse" LA1, "Folgeachse" FA1 und "Kurvenscheibe" KS1 wird ein Gleichlaufverbund hergestellt. Die Verschaltung der Technologischen Objekte erfolgt über Datenflüsse DF1 bis DF3 bzw. DF3'. FIG 4 zeigt die prinzipielle Technologieanordnung für die Realisierung eines Gleichlaufverbundes: Leitwert - Technologisches Objekt "Gleichlauf" GL1 - Technologisches Objekt "Folgeachse" FA1. In FIG 4 wird der Leitwert durch das Technologische Objekt "Leitachse" LA1 repräsentiert. Weiterhin ist in FIG 4 ist dargestellt, dass das Technologische Objekt "Leitachse" LA1 über den Datenflusspfeil DF1 den Leitwert für das Technologische Objekt "Gleichlauf" GL1 vorgibt. Das Technologische Objekt "Leitachse" LA1 kann z.B. eine Positionierachse repräsentieren. Der Leitwert kann aber auch über eine virtuelle Achse d.h. gerechnete (nicht real vorhandene) Achse oder über externe Geber für das Technologische Objekt "Gleichlauf" GL1 vorgegeben werden. Das

Technologische Objekt "Gleichlauf" GL1 stellt als technologische Funktionalität Getriebegleichlauf oder Kurvengleichlauf zur Verfügung, damit können Aufsynchonisieren, Absynchronisieren oder Masterumschaltungen vorgenommen werden. Am Technologischen Objekt "Gleichlauf" GL1 kann als Gleichlaufgesetz wahlweise ein Getriebe oder eine Kurve gewählt werden. Der rechte Teil von FIG 4 stellt diese Auswahlmöglichkeiten dar.

Durch den Zuordnungspfeil ZP1 ist dargestellt, dass der Schalter S1 wahlweise mit einem Getriebe, dargestellt durch den Getriebefaktor GF1 oder mit dem Technologischen Objekt "Kurvenscheibe" KS1 verbunden werden kann. Bei einer Verbindung mit dem Technologischen Objekt "Kurvenscheibe" KS1 erfolgte Datenfluss von diesem Technologischen Objekt zum Technologischen Objekt "Gleichlauf" GL1 über den Datenflusspfeil DF3, den Schalter S1 und den Datenflusspfeil DF3'. Bei einer Verbindung mit dem Getriebefaktor GF1 erfolgt der Datenfluss zum Technologischen Objekt "Gleichlauf" GL1 über den Schalter S1 und den Datenflusspfeil DF3'. Über das Technologische Objekt "Kurvenscheibe" KS1 können nicht lineare Getriebeübersetzungen am Technologischen Objekt "Gleichlauf" GL1 eingestellt werden, über den Getriebefaktor GF1 dagegen lineare Getriebeübersetzungen. Durch den Datenflusspfeil DF2 ist das Technologische Objekt "Gleichlauf" GL1 mit dem Technologischen Objekt "Folgeachse" FA1 verschaltet.

Die Darstellung gemäß FIG 4 zeigt somit die prinzipielle Konfiguration von Technologischen Objekten zur Realisierung einer Gleichlauffunktionalität und kann ihrerseits als (komplexes) Technologisches Objekt angesehen und verwendet werden.

Die Festlegung der Verschaltung der Technologischen Objekte erfolgt in der Konfigurationsphase (Projektierung). Bei Auswahlmöglichkeiten werden diese zur Laufzeit über das Anwenderprogramm (AP1; FIG 2 und AP2, AP3; FIG 11) aktiviert, d.h. zur Laufzeit können Umschaltungen programmiert werden. Prinzipiell können durch Verschaltung mehr als ein "Gleichlaufobjekt" GL1 mit einer "Folgeachse" FA1 verbunden werden, da-

durch wird eine Überlagerung von Gleichlauffunktionen realisiert. Der Leitwert für das "Gleichlaufobjekt" GL1 kann auch direkt aus dem Anwenderprogramm (AP1; FIG 2 und AP2, AP3; FIG 11) vorgegeben werden. Weiterhin kann mehr als ein Technologisches Objekt für die Leitwertbereitstellung konfiguriert werden. Die aktuelle Verschaltung wird wiederum zur Laufzeit über Befehle im Anwenderprogramm (AP1; FIG 2 und AP2, AP3; FIG 11) ausgewählt und aktiviert. Außerdem kann für die Festlegung des Gleichlaufgesetzes zwischen verschiedenen Technologischen Objekten "Kurvenscheibe" KS1 und/oder zwischen verschiedenen Getriebefaktoren GF1 durch Programmierung online umgeschaltet werden. Ein Technologisches Objekt "Kurvenscheibe" KS1 kann einem oder mehreren Technologischen Objekten "Gleichlauf" GL1 zugeordnet werden. Weiterhin können von einem Technologischen Objekt "Leitachse" LA1 eine oder mehrere Gleichlaufverbindungen über Technologische Objekte "Gleichlauf" GL1 konfiguriert werden.

Darstellung gemäß FIG 5 zeigt einen Gleichlaufverbund mit Um-
schaltmöglichkeiten zwischen verschiedenen Leitwertquellen
und Gleichlaufgesetzen, ebenfalls in Form eines Verschal-
tungsdiagramms. In FIG 5 kann das Technologische Objekt
"Gleichlauf" GL2 Leitwerte von den Technologischen Objekten
"Zeit" T, "virtuelle Achse" VA1, "Leitachse" LA2, "Leitachse"
LA3, "externer Geber" EG1 sowie von einem Programmwert TV des
Anwenderprogrammes (AP1; FIG 2 und AP2, AP3; FIG 11) bekom-
men. Durch den Zuordnungspfeil ZP2 ist angedeutet, dass der
Schalter S2 unterschiedliche Leitwertverbindungen für das
Technologische Objekt "Gleichlauf" GL2 herstellen kann. Über
einen der Datenflüsse DF4 bis DF8 sowie über den Schalter S2
und den Datenfluss DF12 wird die „Leitwertverschaltung“ zum
Technologischen Objekt "Gleichlauf" GL2 erreicht. Die Techno-
logischen Objekte "Zeit" T, "virtuelle Achse" VA1, "Leitach-
se" LA2 und LA3, "externer Geber" EG1 sowie der Programmwert
TV sind die potentiellen Master für das Technologische Objekt
"Gleichlauf" GL2. Die möglichen Verschaltungen werden projek-
tiert und die Auswahl eines projektierten Masters kann zur

Laufzeit aus dem Anwenderprogramm (AP1; FIG 2 und AP2, AP3; FIG 11) erfolgen. Damit sind Masterumschaltungen möglich. Das Technologische Objekt "virtuelle Achse" VA1 repräsentiert nicht eine real vorhandene Achse sondern eine gerechnete Achse. "Virtuelle Achsen" sind dadurch gekennzeichnet, dass sie über Befehle kommandiert werden können und eine Bewegungsführung bzw. Interpretation besitzen, aber keine Regelung und keinen Antrieb. Die Technologischen Objekte "Leitachse" LA2 und LA3 repräsentieren dagegen reale Achsen. Reale Achsen 10 repräsentieren Standardachsen mit Antrieb, Motor, Geber, sie besitzen also einen realen Aktor. Auch das Technologische Objekt "externer Geber" EG1 kann einen Leitwert für das Technologische Objekt "Gleichlauf" GL2 bereitstellen. Ein "externer Geber" EG1 besitzt üblicherweise keine Achse und stellt die 15 Informationen in einem projektierbaren Format bereit. "Externe Geber" sind z.B. Winkelgeber an einer Presse. Auch vom Technologischen Objekt "Zeit" T und vom Programmwert TV können Leitwerte für das Technologische Objekt GL2 bereitgestellt werden. Ein Technologisches Objekt "Zeit" stellt einen 20 Leitwert in Form eines Zeitwertes bzw. Zeitfaktors bereit, die Projektierung eines Programmwertes DV als Leitwert erfolgt im Anwenderprogramm (AP1; FIG 2 und AP2, AP3; FIG 11). Die Technologischen Objekte sind hierbei in der üblichen Notation dargestellt.

25 In FIG 5 ist dargestellt, dass als Gleichlaufgesetz für das Technologische Objekt "Gleichlauf" GL2 wahlweise ein Getriebefaktor GF2 oder die Technologischen Objekte "Kurvenscheibe" KS2 und KS3 gewählt werden können. Durch den Zuordnungspfeil ZP3 ist dargestellt, dass der Schalter S3 wahlweise zwischen 30 den Technologischen Objekten KS2, KS3 und dem Getriebefaktor GF2 eingestellt werden kann. Die "getriebemäßige Verschaltung" mit dem Technologischen Objekt "Gleichlauf" GL2 erfolgt dann über die Datenflusspfeile DF9, DF10, den eingestellten 35 Schalter S3 sowie über den Datenflusspfeil DF11. Die Schalteverbindungen S2 und S3 sind im Anwenderprogramm (AP1; FIG 2 und AP2, AP3; FIG 11) programmierbar. Über den Datenfluss-

pfeil DF 13 ist das Technologische Objekt "Gleichlauf" GL2 mit dem Technologischen Objekt "Folgeachse" FA2 verbunden. Das Technologische Objekt "Gleichlauf" GL2 wird bei der Projektierung also slaveseitig mit dem Technologischen Objekt "Folgeachse" FA2 das z.B. eine Gleichlaufachse repräsentieren kann, verschaltet. Masterseitig wird das Technologische Objekt "Gleichlauf" GL2 mit einem Technologischen Objekt verschaltet, dass einen Leitwert zur Verfügung stellt, dieser Leitwert kann auch direkt aus dem Anwenderprogramm (AP1; FIG 2 und AP2, AP3; FIG 11) vorgegeben werden. Somit kann mehr als ein Technologisches Objekt für die Leitwertbereitstellung konfiguriert werden, die aktuelle Verschaltung wird zur Laufzeit über Befehle im Anwenderprogramm ausgewählt.

15 Die Darstellung gemäß FIG 6 zeigt die Verschaltung des Technologischen Objektes "Messtaster" MT1. Die Technologischen Objekte sind hierbei in der üblichen Notation dargestellt. Das Technologische Objekt "Messtaster" MT1 stellt die Funktionalität zur Durchführung eines Messauftrages bereit. Für die 20 Funktionen am Technologischen Objekt "Messtaster" MT1 können Messaufträge aktiviert und parametriert werden. Über den Messeingang ME und den Datenflusspfeil DF14 wird der Messwert an das Technologische Objekt "Messtaster" MT1 geliefert. Der Messeingang ME ist als Ellipse dargestellt. Ein Messeingang ME kann mit mehreren Technologischen Objekten "Messtaster" verschaltet sein. Diese Technologischen Objekte "Messtaster" können dabei auch gleichzeitig aktiviert sein. Ein Messeingang ME entspricht dabei üblicherweise einem Hardware-Messeingang der dem Technologischen Objekt "Messtaster" MT1 25 über Konfiguration zugeordnet wird. Weiterhin ist das Technologische Objekt "Messtaster" MT1 mit mindestens einem Technologischen Objekt verschaltet, das einen Messwert (z.B. Position) liefert. In FIG 7 ist das Technologische Objekt "Messtaster" MT 1 mit den Technologischen Objekten "Achse" A1 und "externer Geber" EG2 über die Datenflusspfeile DF15 bzw. DF16 30 verschaltet. Das Technologische Objekt "Achse" A1 kann z.B. eine Positionierachse oder eine Gleichlaufachse sein. Ein

Technologisches Objekt, das einen Messwert liefert, kann mit mehreren Technologischen Objekten "Messtaster" verschaltet werden.

5 Die Darstellung gemäß FIG 7 zeigt in einem Verschaltungsdia-gramm das Technologische Objekt "Nocke" N1, verschaltet mit den Technologischen Objekten "Achse" A2 und "externer Geber" EG3. Das Technologische Objekt "Achse" A2 ist über den Daten-flusspfeil DF17, das Technologische Objekt "externer Geber" 10 EG3 ist über den Datenflusspfeil DF18 mit dem Technologischen Objekt "Nocke" N1 verschaltet. Über den Datenflusspfeil DF19 ist das Technologische Objekt "Nocke" N1 mit dem Ausgang Out verschaltet, der Ausgang Out ist als Ellipse dargestellt. Das 15 Technologische Objekt "Nocke" N1 stellt die Funktionalität zur Berechnung von Nockenschaltwerten bereit. Über die Funktionen am Technologischen Objekt "Nocke" N1 können Nocken-funktionen aktiviert und parametriert werden. Die Technologi-schen Objekte "Achse" A2 bzw. "externer Geber" EG3 stellen die Bezugswerte für das Technologische Objekt "Nocke" N1 be-reit. Die Zuordnung dieser Technologischen Objekte zum Tech-nologischen Objekt "Nocke" N1 wird vom Anwender projektiert. Der Anwender projektiert weiterhin die Zuordnung des Techno-logischen Objektes "Nocke" N1 zu einem Ausgang Out, dabei ist auch eine Zuordnung auf interne Variablen möglich. Für eine 20 aktuelle Anwendung ist das Technologische Objekt "Nocke" N1 mit genau einem Technologischen Objekt verschaltet, das den Bezugswert liefert.

25

Ein Bezugswert ist z.B. ist eine Achsposition. Hierbei kann 30 das Technologische Objekt "Achse" A2 z.B. eine Positionier-achse oder eine Gleichlaufachse repräsentieren. Dies ist mög-lich, dass eine Zuordnung des Technologischen Objektes "No-cke" N1 auf einen Ausgang Out entfallen kann, dann wirkt das Technologische Objekt "Nocke" N1 nur auf Systemvariablen am 35 Technologischen Objekt (z.B. für die Verwendung des Technolo-gischen Objektes als interne Nocke). Das Technologische Ob-jekt, das den Bezugswert liefert, kann mit mehreren auch un-

terschiedlichen Technologischen Objekten Nocken gleichzeitig verschaltet sein. Die Technologischen Objekte sind hierbei in der üblichen Notation dargestellt.

5 In der Darstellung gemäß FIG 8 wird gezeigt, dass ein Technologisches Objekt "Folgeachse" FA3 mit mehreren Technologischen Objekten "Gleichlauf" GL3 und GL4 verschaltet sein kann. Das Technologische Objekt "Folgeachse" FA3 ist durch den Datenflusspfeil DF22 mit dem Technologischen Objekt
10 "Gleichlauf" GL3 und mit dem Datenflusspfeil DF23 mit dem Technologischen Objekt "Gleichlauf" GL4 verschaltet. Die Technologischen "Gleichlaufobjekte" GL3 und GL4 erhalten über die Datenflusspfeile DF20 bzw. DF21 ihre Leitwertvorgaben. In FIG 8 ist dargestellt, dass die Leitwerte für den jeweiligen
15 Gleichlaufverbund über unterschiedliche Technologische Objekte erfolgen kann. So können für das Gleichlaufobjekt GL3 z.B. das Technologische Objekt "Achse" A3, das Technologische Objekt "virtuelle Achse" VA2 oder das Technologische Objekt "externer Geber" EG4 den Leitwert bereitstellen. Für das
20 "Gleichlaufobjekt" GL4 kann dementsprechend der Leitwert z.B. von den Technologischen Objekten "Achse" A4, "virtuelle Achse" VA3 oder "externer Geber" EG5 bereitgestellt werden. In FIG 8 bilden dann z.B. die Technologischen Objekte "Achse"
25 A4, "Gleichlauf" GL4 und "Folgeachse" FA3 einen Gleichlaufverband. Die jeweils gewünschte Verschaltung wird vom Anwender projektiert, die Auswahl eines projektierten Masters (der Master stellt den Leitwert für den Gleichlaufverbund zur Verfügung) kann zur Laufzeit aus dem Anwenderprogramm erfolgen, damit sind Masterumschaltungen möglich. In FIG 8 stellt das
30 Technologische Objekt "Folgeachse" FA3 den Slave im Gleichlaufverbund dar. Die Technologischen Objekte sind hierbei in der üblichen Notation dargestellt.

Darstellung gemäß FIG 9 zeigt ein Verschaltungsdiagramm bei
35 dem das Technologische Objekt "Kurvenscheibe" KS3 das Getriebegesetz für zwei "Gleichlaufobjekte" GL5 und GL6 über die Datenflusspfeile DF26 bzw. DF27 bereitstellt. In FIG 9 sind

somit zwei Gleichlaufverbunde dargestellt, die jeweils vom gleichen Technologischen Objekt "Kurvenscheibe" KS2 mit einem gemeinsamen Getriebegesetz versorgt werden. Die beiden Gleichlaufverbunde sind links und rechts vom Technologischen 5 Objekt "Kurvenscheibe" KS3 angeordnet. Der linke Gleichlaufverbund wird gebildet durch das Technologische Objekt "Achse" A5, das den Leitwert bereitstellt und somit als Leitachse gilt. Es kann sich dabei z.B. um eine Positionier- oder Gleichlaufachse handeln. Das Technologische Objekt "Achse" A5 10 ist mit dem Datenflusspfeil DF24 mit dem "Gleichlaufobjekt" GL5 verbunden. Über diesen Datenflusspfeil DF24 wird der Leitwert bereitgestellt. Auf der Slaveseite ist das Technologische Objekt "Gleichlauf" GL5 über den Datenflusspfeil GF25 mit dem Technologischen Objekt "Folgeachse" FA4 verbunden. 15 Der rechte Gleichlaufverbund wird gebildet durch die Technologischen Objekte "Achse" A6, "Gleichlauf" GL6 und "Folgeachse" FA5. Die "Achse" A6 entspricht dabei der Leitachse, die "Folgeachse" FA5 repräsentiert eine Slaveachse. Die Verschaltung erfolgt hierbei über die Datenflusspfeile DF28 bzw. 20 DF29. Weiterhin ist es möglich, dass von einer Leichtachse aus eine oder mehrere Gleichlaufverbindungen über Gleichlaufobjekte konfiguriert werden. Technologische Objekte "Kurvenscheibe" können einem oder mehreren Gleichlaufobjekten zugeordnet werden. Die Zusammenstellung von Gleichlaufverbund wird vom Anwender projektiert. Projektierte Gleichlaufverbunde können wiederum als Technologische Objekte repräsentiert 25 werden und ihre Funktionalität in anderen Applikationen wieder verwendet werden. Die Technologischen Objekte sind hierbei in der üblichen Notation dargestellt.

30 Die Darstellung gemäß FIG 10 zeigt die Zusammenfassung von mehreren Technologischen Objekten zu einem Technologiepaket TP. Das Technologiepaket TP ist dabei als Rechteck dargestellt, wobei die linke obere Ecke abgeschnitten ist. Das 35 Technologiepaket TP enthält die Technologischen Objekte "Nocke" N2, "externer Geber" EG6, "Drehzahlachse" DrehA, "Mess-taster" MT2 sowie "Positionierachse" PosA. Die Technologi-

schen Objekte sind dabei in der üblichen Notation dargestellt. Die Technologischen Objekte repräsentieren dabei keine Instanzen, sondern Technologieobjekttypen. Ein Technologiepaket TP enthält somit eine Ansammlung von Technologieobjekttypen, die gewisse Funktionalitäten repräsentieren. Die Zuladung von Technologischen Objekten ins Runtime-System der Steuerung und damit die funktionelle Erweiterung der Steuerung erfolgt über Technologiepakete. Ein Anwender kann sich bestimmte Technologiepakete TP die wiederum Technologieobjekttypen enthalten ins Runtime-System (RTS4, RTS5; FIG 2) laden und somit eine technologische Skalierung der Funktionalität der Steuerung erreichen. Weiterhin kann durch die Technologiepakete TP bei entsprechender Zuordnung von Technologieobjekttypen eine funktionelle Strukturierung erreicht werden.

Darstellung gemäß FIG 11 zeigt in einem Übersichtsbild die Kommunikationsstruktur zwischen zwei Geräten G1 und G2. Gerät bedeutet in diesem Kontext Hardware mit CPU. Die technologische Funktionalität wird in Form von Technologischen Objekten auf Geräte verteilt, auf denen sie letztendlich ablaufen. Softwaretechnisch werden die Geräte G1 und G2 als sog. System-Technologische Objekte (System-TO) dargestellt. Ein System-TO kann nicht verschoben werden, weil es fest an einem Gerät hängt. In einem System-TO ist die Funktionalität des dazugehörigen Gerätes gekapselt. Die System-TOs repräsentieren die Gerätefunktionalität, die Technologischen Objekte die technologische Funktionalität.

Die Geräte G1 und G2 sind in der Darstellung gemäß FIG 11 jeweils als Rechtecke in der linken bzw. rechten Zeichnungshälfte dargestellt. Die Geräte G1 und G2 enthalten jeweils ein Anwenderprogramm AP2 bzw. AP3, TO-Konfigurationen TOK1 bzw. TOK2, technologische Firmware TFW1 bzw. TFW2 und jeweils ein Runtime-System RTS6 bzw. RTS7, wobei alle diese Teilelemente durch Rechtecke dargestellt sind. Die Anwenderprogramme AP2 bzw. AP3 beinhalten die vom Anwender erstellten Befehle

zur Steuerung des technologischen Prozesses (P; FIG1). Bei Bewegungssteuerungen z.B. Positionier- und/oder Bewegungsbefehle. Die technologische Firmware TFW1 bzw. TFW2 stellt die technologische Funktionalität dar, um die das Basissystem (UMC-K; FIG2) der Runtime-Systeme RTS6 bzw. RTS7 erweitert wurde. Die technologische Firmware TWF1 bzw. TWF2 beinhaltet die zugeladenen Technologieobjekttypen, deren Instanzen ein Anwender in seinen Anwenderprogrammen AP2 bzw. AP3 verwenden kann. Die TO-Konfigurationen TOK1 bzw. TOK2 beinhalten Konfigurationsinformationen der Technologischen Objekte (z.B. Verschaltungs- und Verteilungsinformationen). Die Konfigurationen erfolgen im Engineering-System (ES; FIG1). Im Runtime-System RTS6 bzw. RTS7 kommen die Anwenderprogramme letztendlich zum Laufen. Die Runtime-Systeme RTS6 bzw. RTS7 entsprechen einem Betriebssystem und sind z.B. für die Speicherverwaltung und die Rechenzeitverwaltung verantwortlich. Auf die Darstellung weiterer Inhaltselemente der Geräte G1 bzw. G2 wird aus Gründen der Übersichtlichkeit verzichtet.

In der unteren Zeichnungshälfte ist das Kommunikationsmedium KM als langgezogenes Rechteck dargestellt. Das Kommunikationsmedium KM kann z.B. eine Busverbindung darstellen.

Zwischen den Geräten G1 und G2 ist die automatische Kommunikationsprojektierung AKP, ebenfalls als Rechteck dargestellt. Die automatische Kommunikationsprojektierung AKP ist üblicherweise Software, die als Teil des Engineering-Systems (ES; FIG1) abläuft und das Runtime-System RTS6 bzw. RTS7 mit den generierten Kommunikationsinformationen (z.B. wer kommuniziert mit wem? Auf welche Weise erfolgt die Kommunikation?) versorgt.

Der bidirektionale Pfeil LKK zwischen den Anwenderprogrammen AP2 und AP3 stellt einen logischen Kommunikationskanal zwischen den Anwenderprogrammen AP2 und AP3 dar. Der Anwender sieht dabei nur seine Technologischen Objekte, die er selbst

in seinen Anwenderprogrammen verwendet und er kann dabei davon abstrahieren, wo sie physikalisch liegen.

Die gestrichelten unidirektionalen Pfeile DFE1 bis DFE4 stellen den Datenfluss zum Engineering-Zeitpunkt dar. Die automatische Kommunikationsprojektierung AKP wird dabei aus den TO-Konfigurationen TOK1 und TOK2 mit Konfigurationsinformationen zu den technologischen Objekten (z.B. Verteilungs- und Verschaltungsinformationen) über die Datenflüsse DFE1 bzw. DFE2 versorgt. Über die Datenflüsse DFE3 bzw. DFE4 gibt die automatische Kommunikationsprojektierung AKP die daraus generierten Kommunikationskanäle an die Runtime-Systeme RTS6 bzw. RTS7 der Geräte G1 bzw. G2 weiter. Alle Geräte werden somit von der automatischen Kommunikationsprojektierung AKP so mit Routinginformationen versorgt, dass jedes Gerät mit jedem anderen Gerät entsprechend der in den TO-Konfigurationen TOK1 und TOK2 definierten abstrakten Konfigurations- und Kommunikationsbeschreibung kommunizieren kann. Die automatische Kommunikationsprojektierung AKP verwendet zur Generierung der Kommunikationskanäle projektglobale Variablen, mit denen der Anwender z.B. die Qualitätsanforderungen definieren kann.

Die automatische Kommunikationsprojektierung AKP ermöglicht eine effiziente Nutzung der eingesetzten Geräte- und Netztopologie, da sie auch abstrakte Qualitätsanforderungen (z.B. Broadcast, Taktsynchronität, Übertragungszeiten) optimal auf Geräteeigenschaften und Eigenschaften des Kommunikationsmediums KM (z.B. Profibus) abbildet. Bei der Konfiguration der Technologischen Objekte muss sich der Anwender nicht darum kümmern, wie letztendlich die Kommunikation physikalisch stattfindet.

Die gepunkteten vertikalen bidirektionalen Pfeile DKFR1 bis DKFR8 stellen den Daten- und Kontrollfluss zur Laufzeit (Runtime) dar. Z.B. wenn die Geräte G1 bzw. G2 selber am Kommunikationsmedium KM (das Kommunikationsmedium kann z.B. ein Profibus sein) hängen, anlaufen und in Betrieb sind. Dann er-

folgt nämlich wirklich ein "scharfer" Daten- und Kontrollfluss von den Anwenderprogrammen AP2 bzw. AP3 durch die technologische Firmware TFW1 bzw. TFW2 durch das Runtime-System RTS6 bzw. RTS7 auf das Kommunikationsmedium KM, über das Kommunikationsmedium KM zum nächsten Gerät und da wieder nach oben zum Anwenderprogramm. Im "scharfen Betrieb" eines Gerätes werden natürlich auch die Informationen der TO-Konfiguration TOK1 bzw. TOK2 benötigt.

Patentansprüche

1. Industrielle Steuerung für technische Prozesse, insbesondere für Produktionsmaschinen,

5 d a d u r c h g e k e n n z e i c h n e t,
dass die Steuerung ein allgemein einsetzbares, vorzugsweise technologieneutrales Basissystem (UMC-K) für die Steuerungsgrundfunktionalität aufweist, wobei instanzierbare Technologieobjekttypen die Grundfunktionalität der Steuerung um technologische Funktionalitäten ergänzen und nach einer vom Anwender zuschneidbaren Instanziierung als Technologische Objekte (TO1 - TOn) in seinen jeweiligen Applikationen zur Verfügung stehen, wobei eine Trennung zwischen technologischer Funktionalität und Gerätefunktionalität erfolgt.

15

2. Industrielle Steuerung nach Anspruch 1,

 d a d u r c h g e k e n n z e i c h n e t,
dass eine automatische Generierung bzw. Projektierung von Kommunikationsverbindungen zwischen Technologischen Objekten
20 (TO1 - TOn) basierend auf der zugrundeliegenden Hardware-Topologie und/oder der technologischen Lösung erfolgt.

3. Industrielle Steuerung nach Anspruch 1 oder 2,

 d a d u r c h g e k e n n z e i c h n e t,
25 dass bei der automatischen Generierung bzw. Projektierung der Kommunikationsverbindungen zwischen Technologischen Objekten (TO1 - TOn) von den Technologischen Objekten (TO1 - TOn) zu- gewiesene oder erworbene Qualitätsattribute berücksichtigt werden.

30

4. Industrielle Steuerung nach Anspruch 1, 2 oder 3,

 d a d u r c h g e k e n n z e i c h n e t,
dass eine flexible Verschiebbarkeit und/oder Verteilbarkeit
35 der Technologischen Objekte (TO1 - TOn) auf unterschiedlich oder gleich performante Hardware-Systeme und/oder Laufzeit- systeme (RTS1 - RTS7) erfolgt.

5. Industrielle Steuerung nach Anspruch 4,
dadurch gekennzeichnet,
dass eine flexible Verschiebbarkeit und/oder Verteilbarkeit
der Technologischen Objekte (TO1 - TOn) auf unterschiedlich
5 oder gleich performante Hardware-Systeme und/oder Laufzeit-
systeme (RTS1 - RTS7) innerhalb eines Projektes erfolgt, wo-
bei sich ein Projekt auf Daten und/oder Programme von einer
oder mehreren Steuerungseinheiten bezieht.

10 6. Industrielle Steuerung nach einem der vorstehenden Ansprü-
che,
dadurch gekennzeichnet,
dass die Verteilung der Funktionalität der Technologischen
Objekte (TO1 - TOn) auf miteinander in Echtzeit durch takt-
15 synchron äquidistant kommunizierende Steuerungseinheiten er-
folgt.

7. Industrielle Steuerung nach einem der vorstehenden Ansprü-
che,
20 dadurch gekennzeichnet,
dass eine technologische Skalierung hinsichtlich der Funktio-
nalität der Steuerung durch die Zuladbarkeit von Technologie-
objekttypen erfolgt.

25 8. Industrielle Steuerung nach einem der vorstehenden Ansprü-
che,
dadurch gekennzeichnet,
dass eine Verschaltung der Technologischen Objekte (TO1 -
TON) zu komplexen Technologischen Objekten, sog. Container-
30 Objekten erfolgt.

9. Industrielle Steuerung nach einem der vorstehenden Ansprü-
che,
dadurch gekennzeichnet,
35 dass einem Anwender unterschiedliche Sichten auf die Techno-
logischen Objekte (TO1 - TOn) zur Verfügung stehen.

10. Industrielle Steuerung nach einem der vorstehenden Ansprüche,

d a d u r c h g e k e n n z e i c h n e t,

dass eine rückwirkungsfreie Programmierung eines Technologischen Objektes (TO1 - TOn) bezüglich der anderen vorhandenen Technologischen Objekte und des Steuerungsbasisystems (UMC-K), sofern nicht explizit eine Rückwirkung programmiert bzw. projektiert ist, vorgesehen ist.

10 11. Industrielle Steuerung nach einem der vorstehenden Ansprüche,

d a d u r c h g e k e n n z e i c h n e t,

dass die Darstellung der Technologischen Objekte (TO1 - TOn) im Engineering-System (ES) durch grafische Elemente und/oder Masken erfolgt.

12. Industrielle Steuerung nach einem der vorstehenden Ansprüche,

d a d u r c h g e k e n n z e i c h n e t,

20 dass die Technologieobjekttypen zu Technologischen Paketen (TP) zusammengefasst werden.

13. Verfahren zur Programmierung bzw. Projektierung von industriellen Steuerungen für technische Prozesse, insbesondere für Produktionsmaschinen,

25 g e k e n n z e i c h n e t d u r c h die Verwendung von Technologischen Objekten (TO1 - TOn) und die Abfolge der folgenden Schritte:

a) Verwendung eines Basissystem (UMC-K) mit einer vorzugsweise technologieneutralen Grundfunktionalität,

30 b) Instanziierung der Technologischen Objekte (TO1 - TOn),

c) Verschaltung der Technologischen Objekte (TO1 - TOn) zu Technologischen Objekten komplexer Funktionalität,

d) Verteilung und/oder Platzierung der Technologischen Objekte (TO1 - TOn) auf die Geräte (G1, G2),

35 e) automatische Generierung der Kommunikationskanäle zwischen den Technologischen Objekten (TO1 - TOn),

f) Wiederverwendung insbesondere von komplexen bereits verschalteten Technologischen Objekten in anderen Projekten.

14. Verfahren zur Programmierung bzw. Projektierung nach An-
spruch 13,

d a d u r c h g e k e n n z e i c h n e t,
dass bei der Generierung der Kommunikationskanäle Qualitäts-
attribute der Technologischen Objekte (TO1 - TOn) eingehalten
werden.

10

15. Verfahren zur Programmierung bzw. Projektierung nach An-
spruch 13,

d a d u r c h g e k e n n z e i c h n e t,
dass die Schritte b) und e) optional erfolgen.

Zusammenfassung

Industrielle Steuerung auf der Basis verteilbarer Technologischer Objekte

5

Durch die Zuladbarkeit von Technologischen Objekttypen (TO1-TOn) in das Runtime-System (RTS1 - RTS7) einer industriellen Steuerung wird das Basissystem (UMC-K) der Steuerung funktional erweitert und eine technologische Skalierung der Steuerung erreicht. Die zugeladenen Technologischen Objekttypen sind beliebig instanzierbar und können beliebig auf die vorhandenen Geräte verteilt werden. Das Zuladen erfolgt in Form von Technologiepaketen (TP). Der Anwender kann in seinen Anwenderprogrammen (AP1 - AP3) diese Funktionalität direkt verwenden, wobei eine Trennung zwischen technologischer Funktionalität und Gerätefunktionalität erfolgt.

10
15
FIG 11

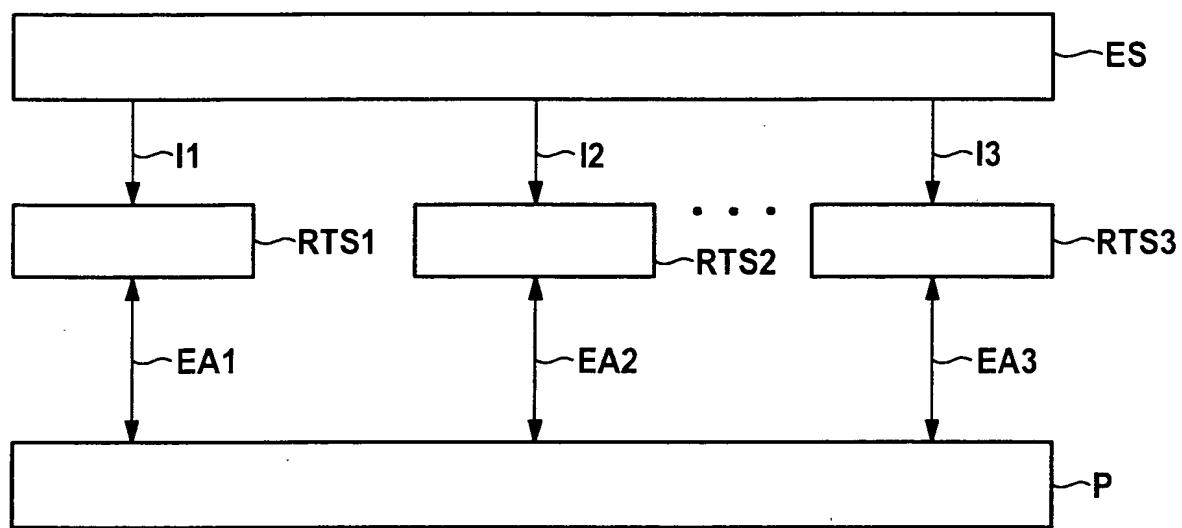


FIG 1

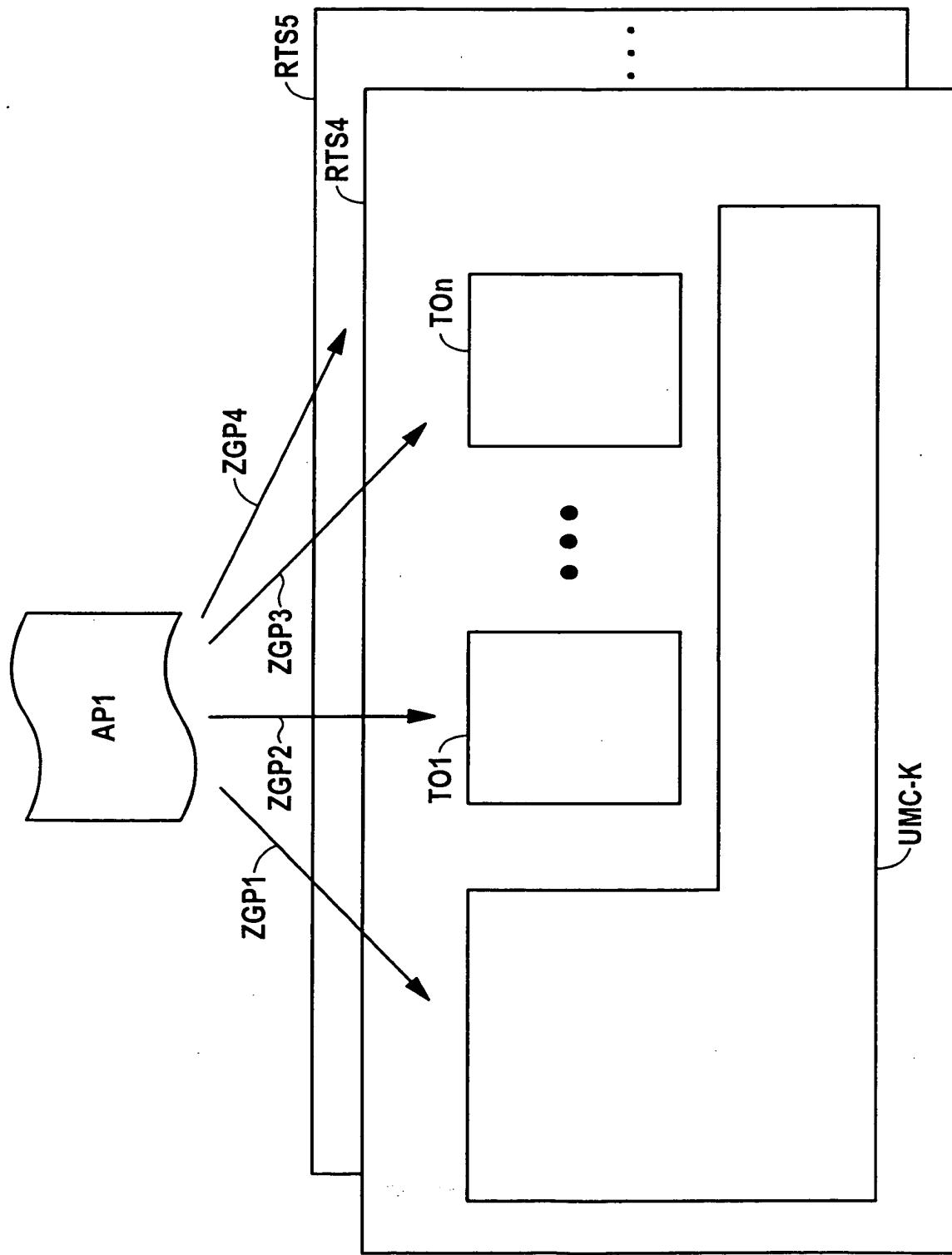


FIG 2

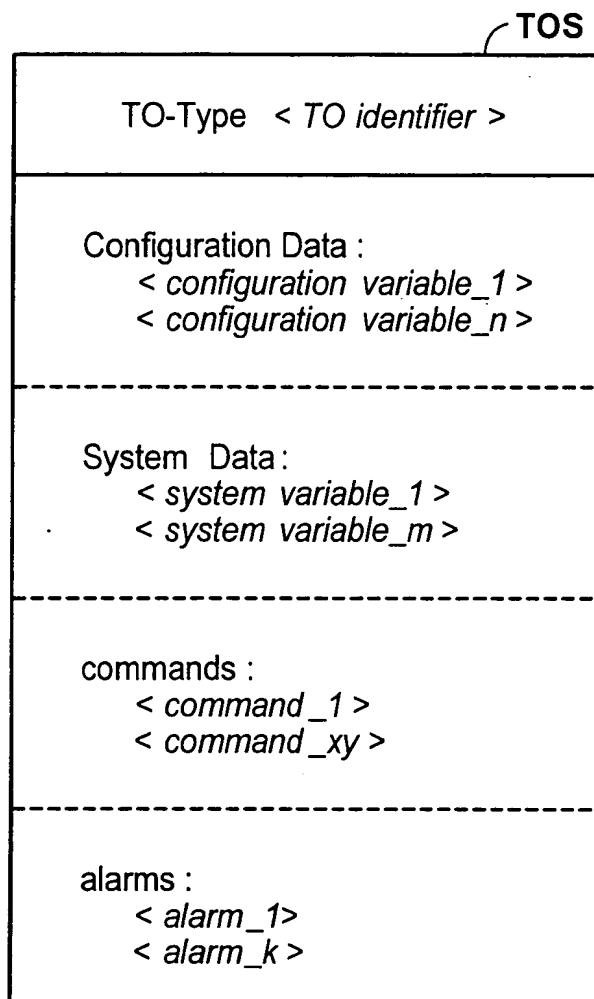


FIG 3

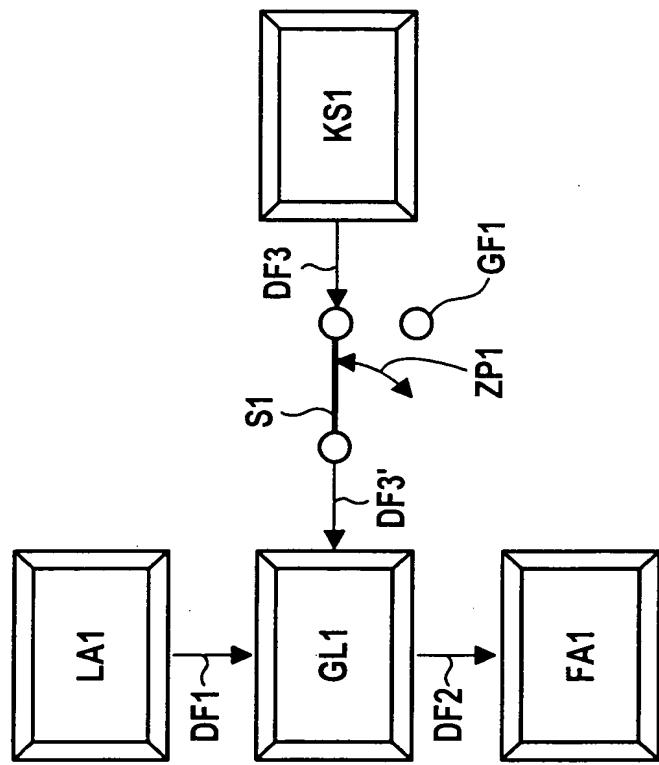


FIG 4

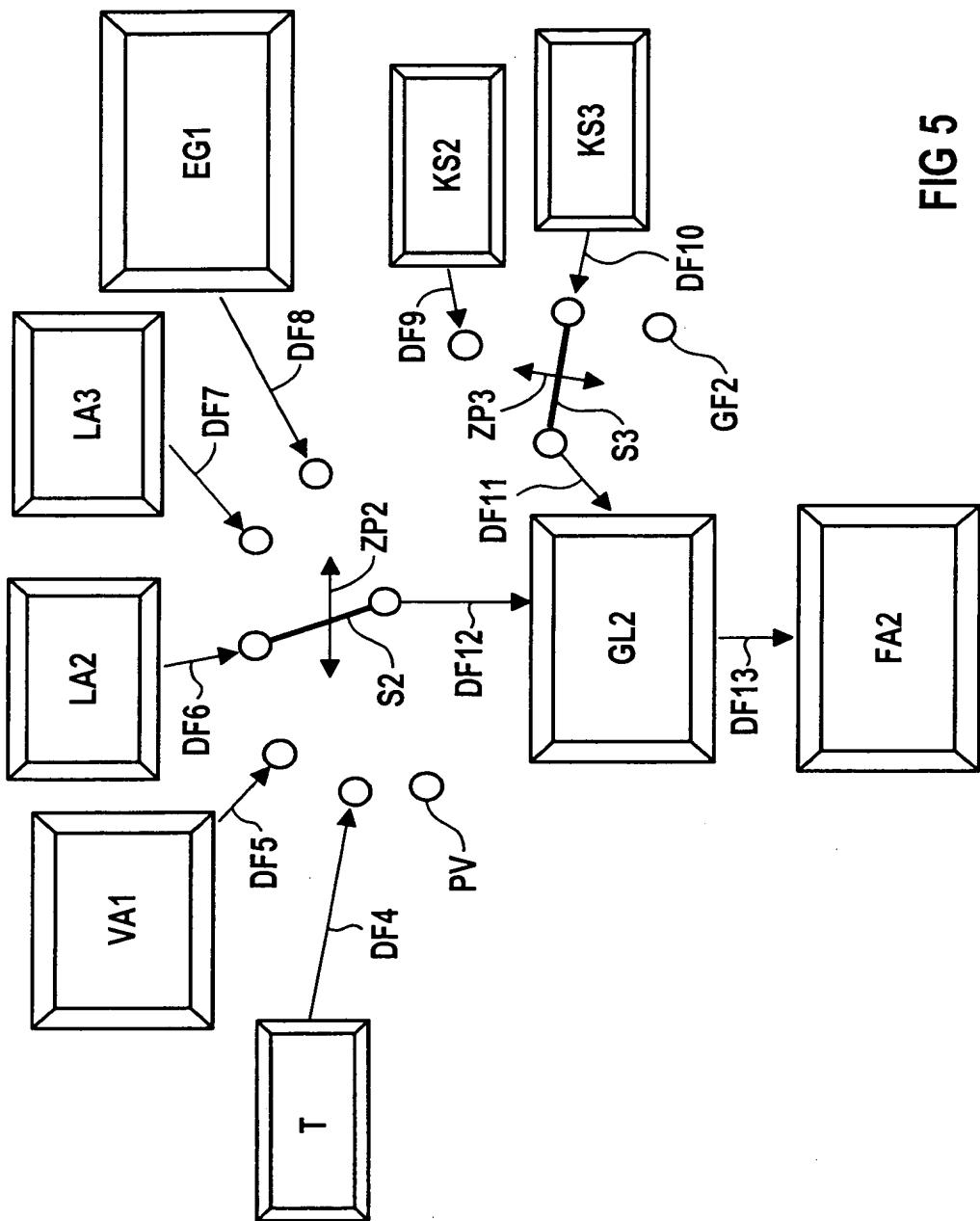


FIG 5

6/11

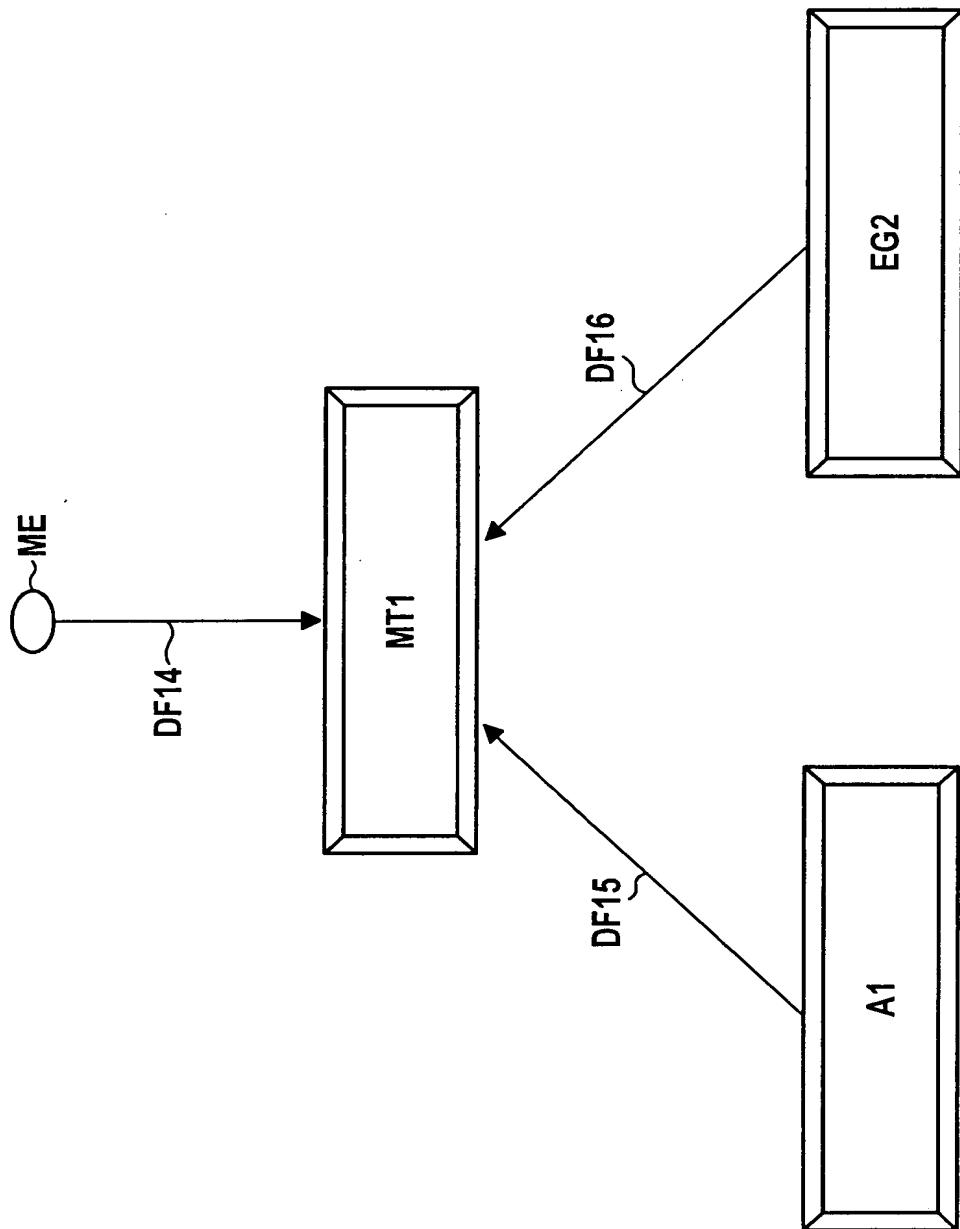
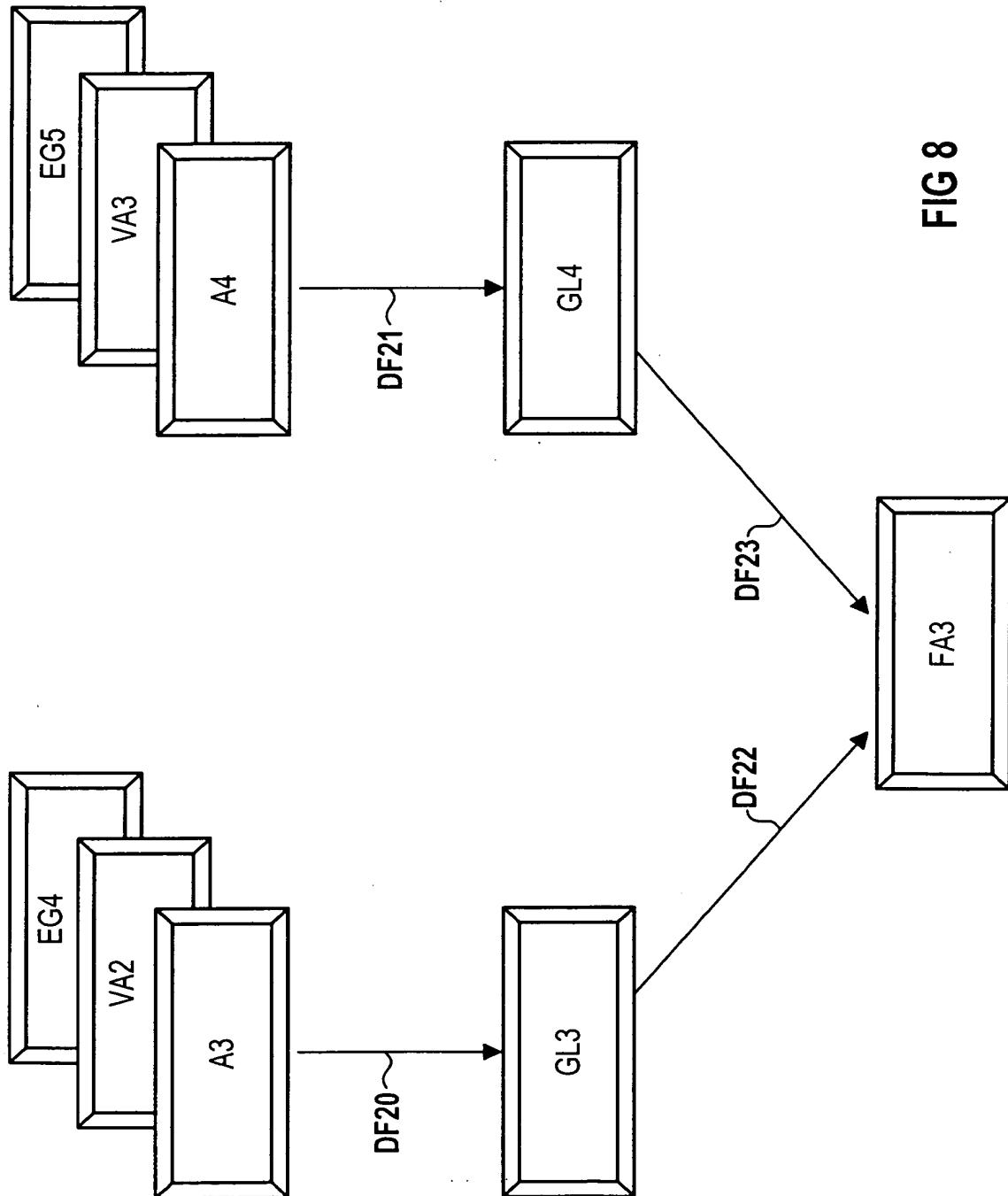


FIG 6



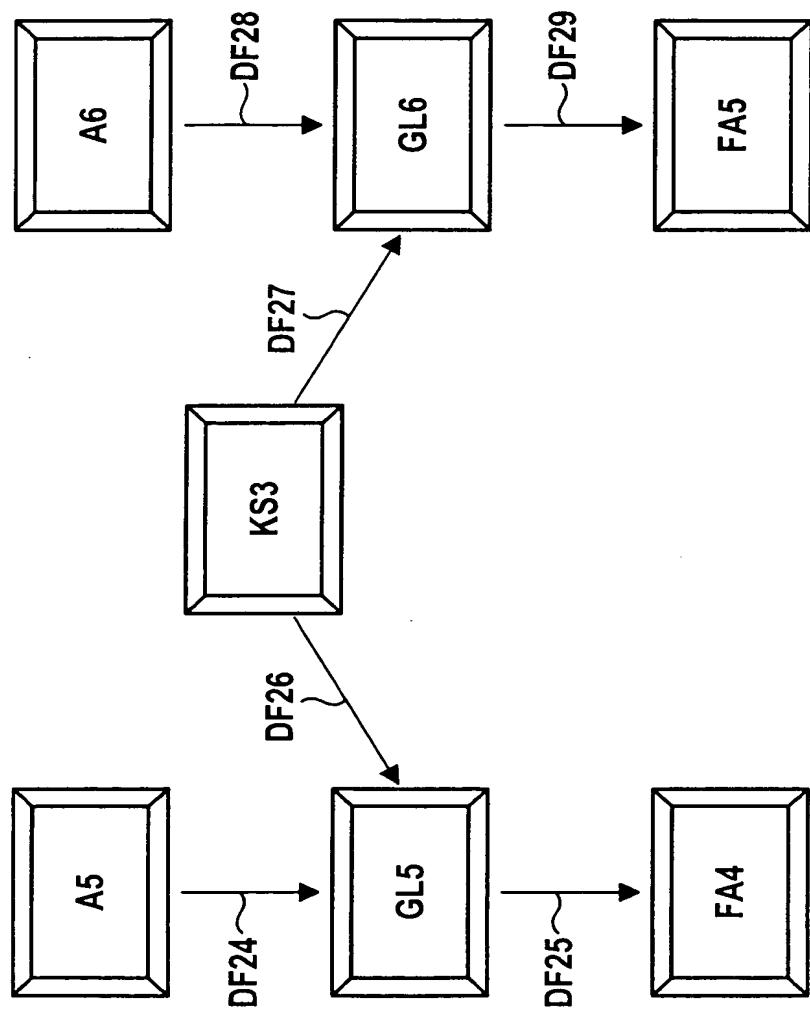


FIG 9

10/11

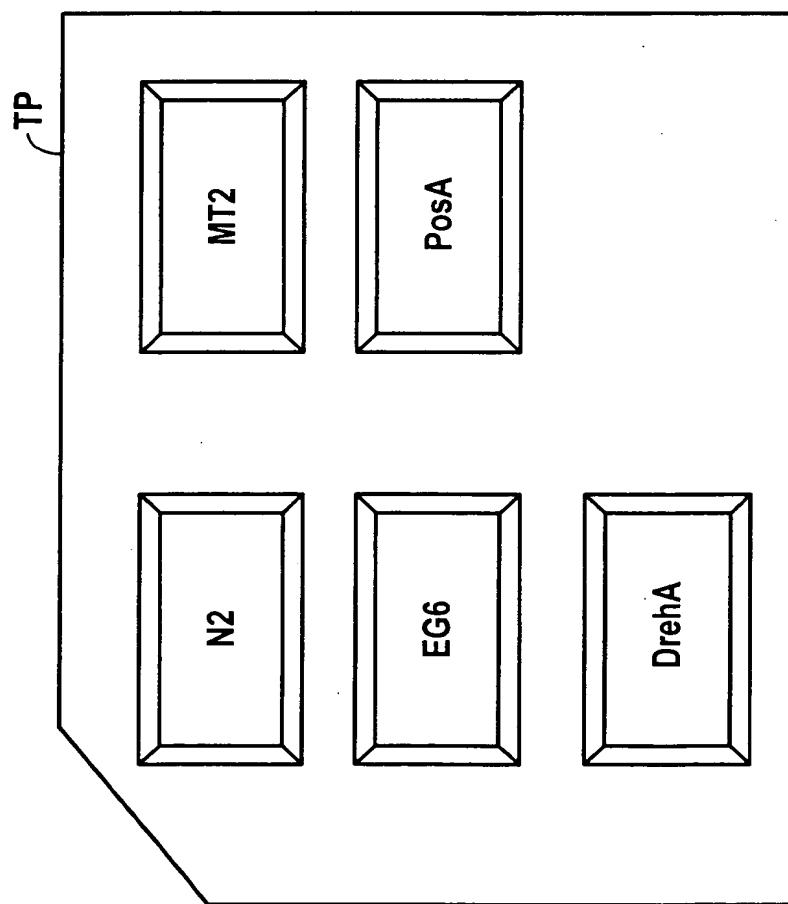


FIG 10

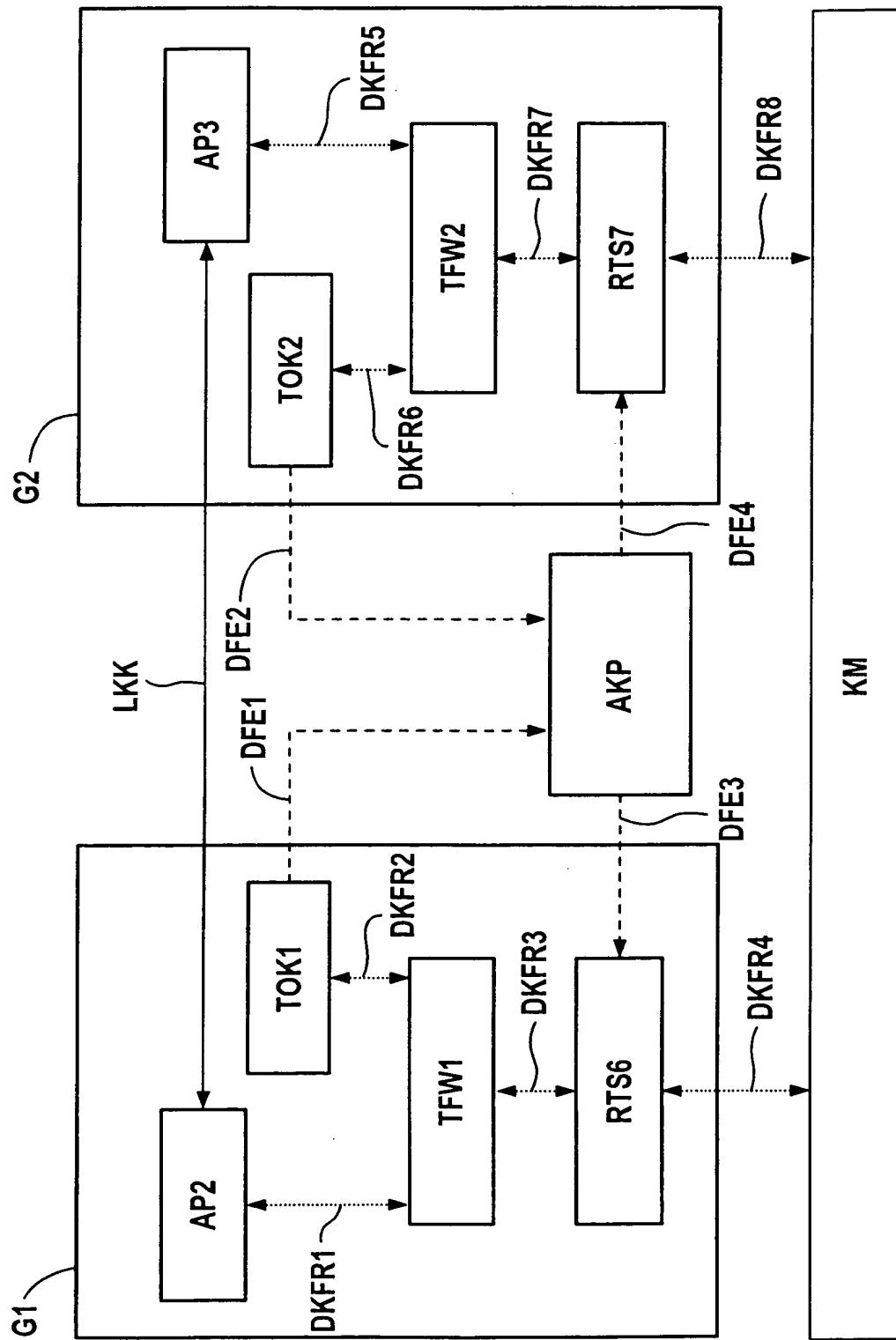


FIG 11